

Apuntes para visualizar datos en mapas con R

con `tm_shape()`

@eloimm

2019-01-22

El Shapefile (.shp) del territorio

Cargamos poligonos del shapefile

```
library(rgdal)
poligons<- readOGR("data/Comarques/shapefiles_catalunya_comarcas.shp",
                  "shapefiles_catalunya_comarcas",
                  use_iconv = TRUE, # Sin este paramtro da errores en los accents para el catalan
                  encoding = "UTF-8")
```

Para visualizar datos contenidos en el archivo

```
head(poligons@data)
poligons@data$nom_comar
```

Podemos operar en ellos por ejemplo para *actualizar* la denominación de la “Val d’Aran” por Aran

```
library(tidyverse)
poligons@data$nom_comar <- as.character(poligons@data$nom_comar)
poligons@data$nom_comar[poligons@data$nom_comar=="Val d'Aran"] <- "Aran"
poligons@data$nom_comar <- sort(poligons@data$nom_comar)
poligons@data$nom_comar <- as.factor(poligons@data$nom_comar)
```

Los datos

Cargamos los datos que queremos representar

```
load(file = "data/dades.RData")
```

`merge()`

Creamos una columna en los datos comun con los polígonos

```
dades$id <- poligons@data$comarca
```

Ahora podemos juntos los dos objetos

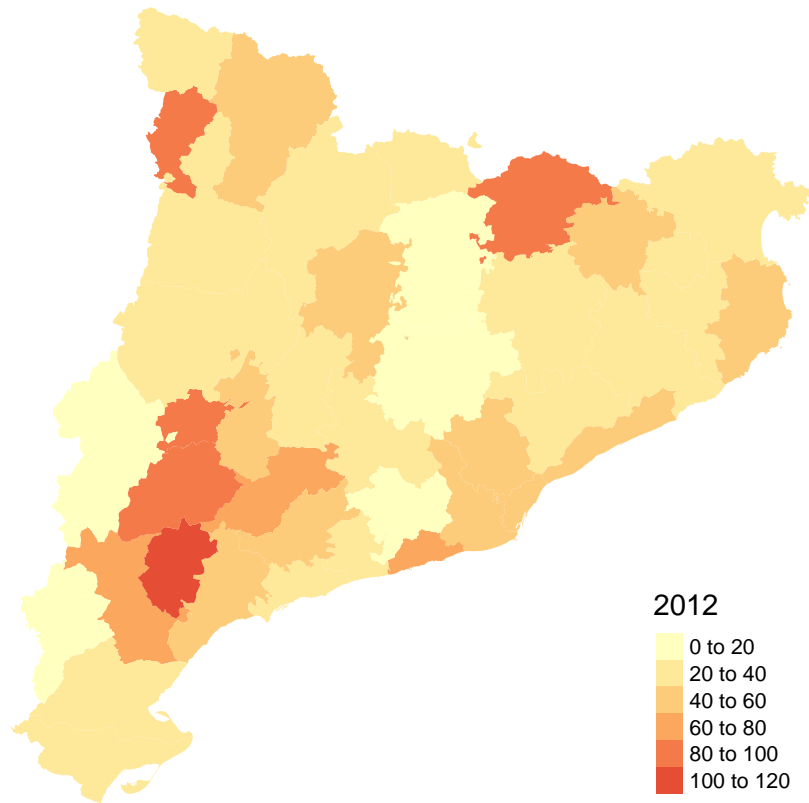
```
mapa <- merge(poligons, dades, by.x = "comarca", by.y= "id")
```

Dibujamos los mapas

```
library(tmap)
mp1 <- tm_shape(mapa) +
  tm_fill("2012", # Columna a pintar
          palette = "-RdYlGn")+
```

```
tm_layout(frame = FALSE)
```

mp1



Guardamos en formato .png

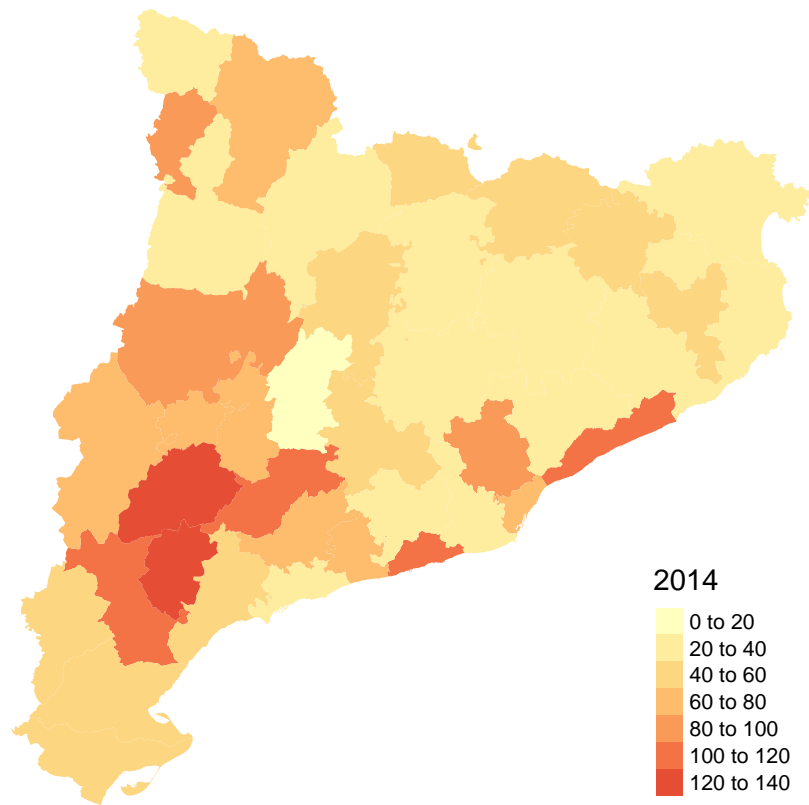
```
png(filename = "images/2012.png")
mp1
dev.off()
```

Retornamos a **imprimir** mapas:

```
tmap_mode("plot")
```

Si queremos compara el mapa de 2012 con el del año 2014...

```
# Creamos el mapa 2014
mp2 <- tm_shape(mapa) +
  tm_fill("2014", # Columna a pintar
    palette = "-RdYlGn")+
  tm_layout(frame = FALSE)
mp2
```

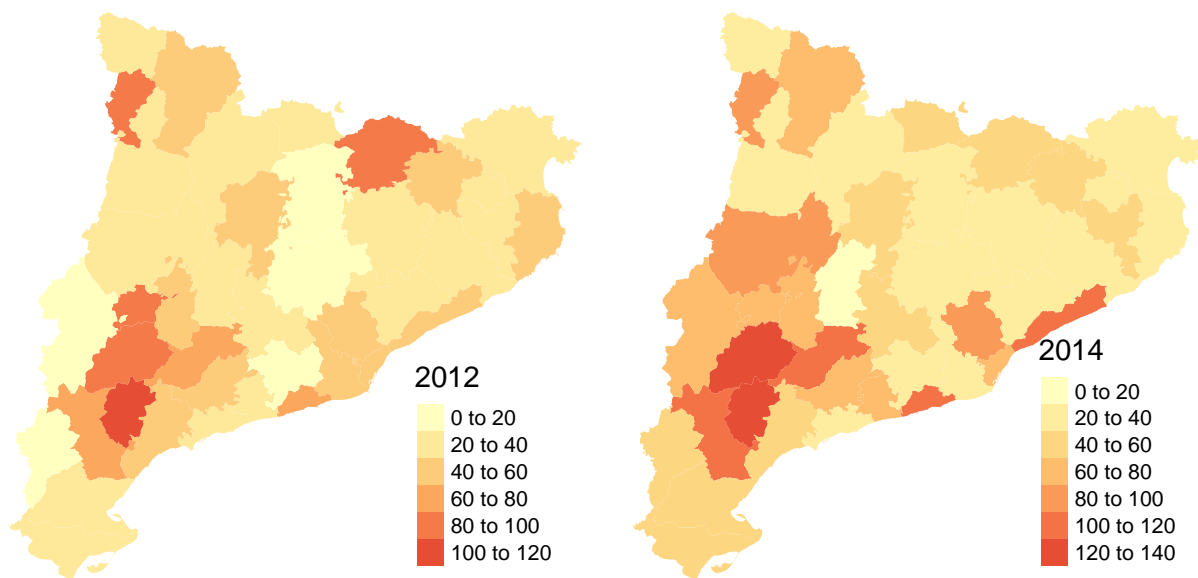


```
# Guardamos el mapa 2014
png(filename = "images/2014.png")
mp2
dev.off()
```

Los plotamos uno junto al lado del otro

```
library(grid)
library(gridExtra)
library(ggplot2)

grid.newpage()
pushViewport(viewport(layout=grid.layout(1,2)))
print(mp1, vp=viewport(layout.pos.col = 1, layout.pos.row =1))
print(mp2, vp=viewport(layout.pos.col = 2, layout.pos.row =1))
```



```
# Para guardarlo como .png
png(filename = "images/Mp1Mp2.png")
grid.newpage()
pushViewport(viewport(layout=grid.layout(1,2)))
print(mp1, vp=viewport(layout.pos.col = 1, layout.pos.row =1))
print(mp2, vp=viewport(layout.pos.col = 2, layout.pos.row =1))
dev.off()
```

Los valores y colores de la escala nos distorcian poder comparar los mapas.

Para solucionarlo debemos asignar los rangos y *fijar* el estilo para cada año

```
max(mapa$`2012`)
max(mapa$`2013`)
max(mapa$`2014`)
#El valor máximo de los valores es 150.1. Creamos un rango de valores hasta 160
breaks <- c(20,40,60,80,100, 120, 140,160)

map1 <- tm_shape(mapa) +
  tm_fill("2012",
    breaks = breaks, palette = "-RdYlGn", style = "fixed")+
  tm_layout(legend.show = FALSE, frame = FALSE)

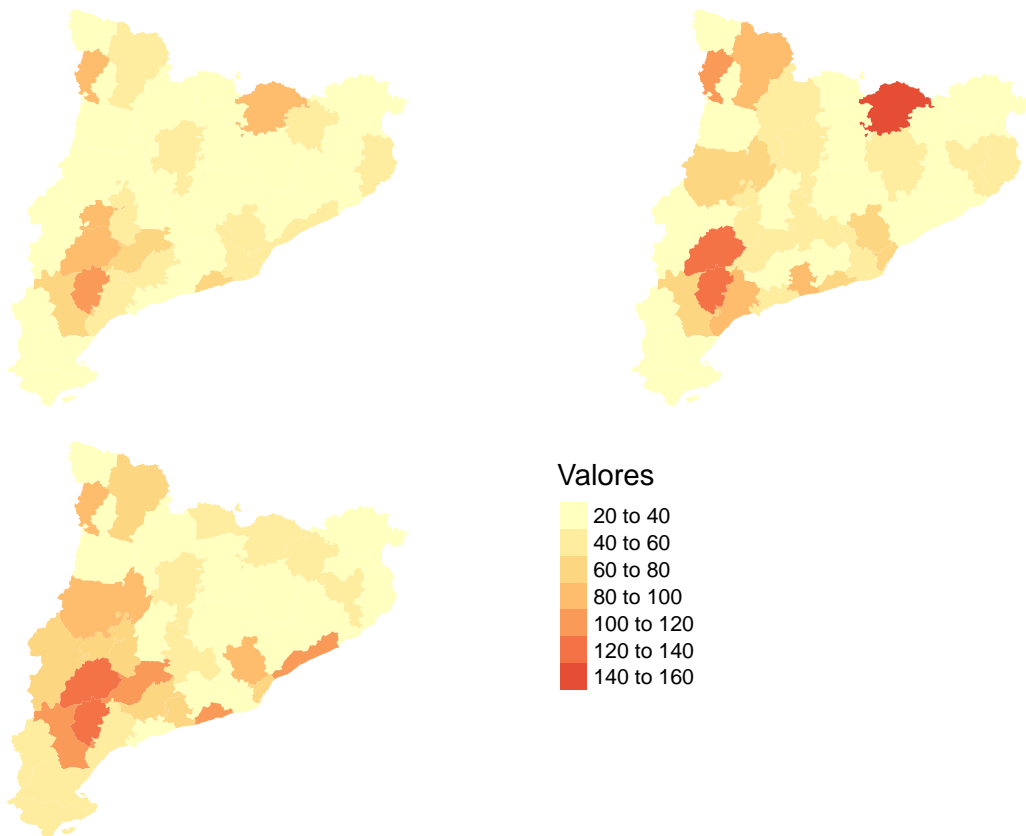
map2 <-tm_shape(mapa) +
  tm_fill("2013", breaks = breaks, palette = "-RdYlGn", style = "fixed")+
  tm_layout(legend.show = FALSE, frame = FALSE)
```

```
map3 <- tm_shape(mapa) +
  tm_fill("2014", breaks = breaks, palette = "-RdYlGn", style = "fixed")+
  tm_layout(legend.show = FALSE, frame = FALSE, legend.outside=TRUE, legend.outside.position="bottom")

# Tomamos los valores (p.e del mapa 2014) y los guardamos para la leyenda del mapa
legend.map <- tm_shape(mapa) +
  tm_fill("2014", title = "Valores",
    breaks = breaks,
    palette = "-RdYlGn", style = "fixed") +
  tm_layout(legend.only = TRUE)
```

Imprimimos y guardamos el mapa como png

```
grid.newpage()
pushViewport(viewport(layout=grid.layout(2,2)))
print(map1, vp=viewport(layout.pos.col = 1, layout.pos.row =1))
print(map2, vp=viewport(layout.pos.col = 2, layout.pos.row =1))
print(map3, vp=viewport(layout.pos.col = 1, layout.pos.row =2))
print(legend.map, vp=viewport(layout.pos.row = 2, layout.pos.col = 2))
```



```
png(filename = "images/Map1Map2Map3Leg.png")
grid.newpage()
pushViewport(viewport(layout=grid.layout(2,2)))
print(map1, vp=viewport(layout.pos.col = 1, layout.pos.row =1))
print(map2, vp=viewport(layout.pos.col = 2, layout.pos.row =1))
print(map3, vp=viewport(layout.pos.col = 1, layout.pos.row =2))
print(legend.map, vp=viewport(layout.pos.row = 2, layout.pos.col = 2))
```

```
dev.off()
```

Para más información:

- Guy Lansley and James Cheshire (2016) [An Introduction to Spatial Data Analysis and Visualisation in R](#)
- Robin Lovelace, Jakub Nowosad, Jannes Muenchow (2019) [Geocomputation with R](#)